

## Problem Statement

You're at a music festival with **S stages** and **P performances** throughout the day. Each performance has a dynamic "hype level" that changes over time, and you want to maximize your total satisfaction!

### Festival Mechanics:

- Each performance happens at a specific stage, with start and end times
- Hype level for a performance =  $\text{base\_hype} + (\text{minutes\_since\_start} \times \text{hype\_growth})$
- Your satisfaction from a performance =  $\text{hype\_level} \times \text{minutes\_watched}$
- Walking between stages takes time (given as a distance matrix)
- You have limited energy: can only change stages **K times**
- **Combo bonus:** Certain performance pairs give bonus satisfaction if seen in sequence

**Objective:** Plan your festival route to maximize total satisfaction points.

### Input Format

```
S P K
[S×S distance matrix - minutes to walk between stages]
[P performance lines: stage start_time end_time base_hype hype_growth]
C
[C combo lines: performance1_id performance2_id bonus_points]
```

### Output Format

```
Maximum satisfaction points achievable
Schedule: [performance_id: arrival_time-departure_time]
```

---

### Sample Input

```
2 3 1
0 3
3 0
0 0 40 5 3
0 45 80 10 2
1 30 60 30 1
0
```

### Sample Output

```
1740
Schedule: [P0: 0-40, P2: 43-60]
```

**Explanation:**

- Watch all of P0 (40 minutes at stage 0)
  - Walk to stage 1 (3 minutes)
  - Watch end of P2 (17 minutes)
  - Only used 1 stage change (within limit)
- 

### Challenge Input

```

3 4 2
0 5 10
5 0 5
10 5 0
0 0 30 10 2
1 20 50 15 1
1 60 90 20 3
2 40 70 25 1
1
1 3 50

```

---

### Concepts to Discuss

- How do you model the decision space?
  - Greedy vs. dynamic programming approaches?
  - How do combo bonuses affect your algorithm?
  - What's the trade-off between staying longer vs. seeing more performances?
  - How does the energy limit (K stage changes) constrain your solution?
- 

### Tips for Teams

- Start by understanding the problem with small examples
- Draw diagrams and trace through sample inputs by hand
- Consider multiple approaches before committing to one
- Think about edge cases and constraints
- Present your solution clearly with pseudocode
- Discuss time/space complexity of your approach
- Have fun and think creatively!